THE QL MAGAZINE

Volume 31 Issue 5 AUG-SEP

BASIC is 50 - Dartmouth

QL is 30 - Edinburgh

NEMQLUG is 21 - Manchester

QUANTA
QL USERS & TINKERERS ASSOCIATION

QUANTA

INFORMATION ON THE ASSOCIATION

Membership of QUANTA, the independent QL user group, is by annual subscription. The Membership Secretary can supply full details. Copies of the association's constitution & annual accounts are available from the Secretary.

QUANTA Committee - 2014/2015

Chairman: Sarah D. Gilpin 181, Urmston Lane, Stretford, Manchester. M32 9EH
Email: chairman@quanta.org.uk Tel: 0161 865 2872

Secretary: Alison Southern 40, Distaff Road, Poynton, Cheshire. SK12 1HN
Email: secretary@quanta.org.uk Tel: 01625 850067

Treasurer & Membership Secretary: John Gilpin 181, Urmston Lane, Stretford, Manchester. M32 9EH
Email: treasurer@quanta.org.uk
Email: membership@quanta.org.uk Tel: 0161 865 2872

Magazine Editor: Lee Privett. 25 Mucking Hall Road, Barling Magna, Southend on Sea, Essex. SS3 0NH.
Email: editor@quanta.org.uk Mobile: 07985 513 234

Librarian: David Buckley 136, Denton Lane, Chadderton, Oldham. OL9 8PS
Email: librarian@quanta.org.uk Tel: 0161 622 1600

Helpline & News Editor: Dilwyn Jones 22 Erw Las, Coetmor New Road, Bethesda, Gwynedd. LL57 3NN
Email: helpline@quanta.org.uk
Email: news@quanta.org.uk Tel: 01248 601599

Web Administrator & Web Master: Keith Dunbar 44, Dalton Avenue, Stretford, Manchester. M32 9TP
Email: webadmin@quanta.org.uk Mobile: 07789 483 800

I.C.M.: Dave Park, 1406 Dana Court, Round Rock, TX USA 78664
Email: dave@quanta.org.uk Tel: (512) 529 7863

Back Issues Requests: Basil Lee 31 Fairdale Gardens, London. SW15 6JW
Email: backissues@quanta.org.uk Tel: 0208 789 1976

QUANTA reserves the right to publish, reject or edit any material submitted. The opinions expressed are solely those of the contributors. Member's private small ads - free (up to 50 words). Non-members private small ads (and members after 50 words) 50p per 10 words or part thereof. Terms strictly paid up-front with order. Receipt if requested. All copyrights and trademarks are hereby acknowledged.

CONTENTS

Editorial .... Lee Privett 4
QUANTA News .... Dilwyn Jones 5
Quo Vadis Design .... Advertisement 8
RWAP Software .... Advertisement 16
Chairman's Notes................. Sarah Gilpin......................20
Parameter Lists....................George Gwilt 21
Jochen Merz Software........ Advertisement 22
QUANTA Helpline................. Dilwyn Jones 32
RWAP Membranes .... Advertisement 40
NEMQLUG Sub-Group News .. Sarah Gilpin......................49
QL is 30 workshop .... Geoff Wicks......................50
Comment .... Lee Privett.......................52
Late News .... Dilwyn Jones 53
Copy Date .... (For Next Issue) 55
QL Forum .... Advertisement..................56

Visit our website at : www.QUANTA.org.uk
'Like' us on Facebook at : www.Facebook.com/QUANTA.org

Currently watching this video? https://www.youtube.com/watch?v=AO5BUIKykMM

Just back from my honeymoon, this is the excuse (which is really true) I am giving for the slight delay in getting this issue to you. Now, the flavour of the front cover, if you haven't already guessed in this ‘QL is 30’ special year takes another homage from a QL World magazine front cover March 1992. We hope you like it.

The summer issue is, as in past years, a light one with regards to content, however, we have yet another interesting article from George Gwilt covering “Parameter Lists” with the code that still baffles me. Perhaps someone can explain the basics for a future issue. We always can use content for articles reviews etc.

Without support from members it become increasingly hard to fill the magazine each year with content. So yet another plea from me regarding copy goes out to you in this issue.

Currently our list request from members is growing, however the number of new responses are deafeningly quiet, at just under 1, YES zero.

A reminder please:

We are seeking a volunteer for the role of Treasurer
We are seeking a volunteer for the role of Membership Secretary
Or we are seeking a volunteer for the role of both (above)
We would like feedback about your attendance at Edinburgh, QL is 30
We are seeking feedback regarding e-mag (see last issue)
We are seeking feedback on anything that you do
We are seeking feedback on anything we do

The Editor

If you have QL-related news items that you'd like us to include on this page, please get in touch with News Editor - Dilwyn Jones at news@quanta.org.uk

This month's News column shows it's been a summer with a focus on games for the QL. After getting hooked on SoQLoban mentioned in the last issue, I've been reluctant to play these games too much for fear of making this issue late, but hopefully you'll have great fun with all the games if the weather is miserable in the near future!

txtElite

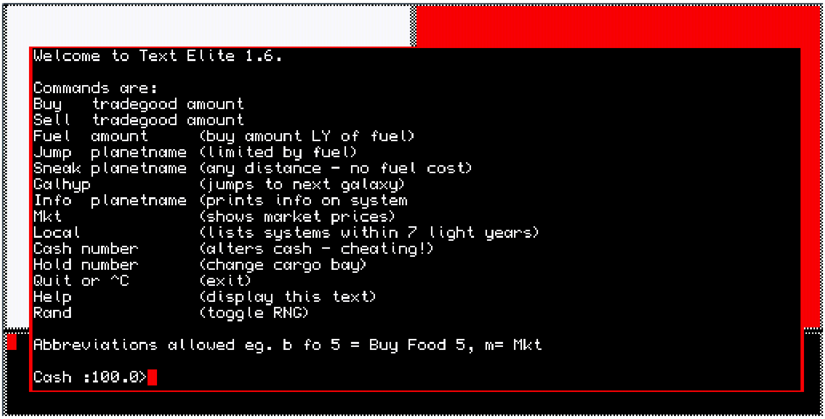
Derek Stewart has successfully ported the txtElite version of the space trading game Elite to the QL.

Elite became popular as a 3D graphics game on systems such as the BBC micro in the mid 1980s and since then has been ported to all sorts of systems. It was originally written by Ian Bell and David Braben.

txtElite is the space trading part of the original Elite, minus the graphics (apparently an Elite graphics demo was once done for a QL, which I am currently looking for).

As part of the port, Derek has improved the screen input routines, as the text version relies so much on user input.
A brief text file is included with the archive when you download it, and for a more general and in-depth look at Elite, have a look at an online manual on Ian Bell's website at:
http://www.iancgbell.clara.net/elite/manual.htm

txtElite for QL may be downloaded from the Games page on my website at http://www.dilwyn.me.uk/games/index.html



```
Welcome to Text Elite 1.6.

Commands are:
Buy   tradegood amount
Sell  tradegood amount
Fuel  amount      (buy amount LY of fuel)
Jump  planetname  (limited by fuel)
Sneak planetname  (any distance - no fuel cost)
Galhyp            (jumps to next galaxy)
Info  planetname  (prints info on system
Mkt               (shows market prices)
Local             (lists systems within 7 light years)
Cash number       (alters cash - cheating!)
Hold number       (change cargo bay)
Quit or ^C        (exit)
Help              (display this text)
Rand              (toggle RNG)

Abbreviations allowed eg. b fo 5 = Buy Food 5, m= Mkt

Cash :100.0>
```

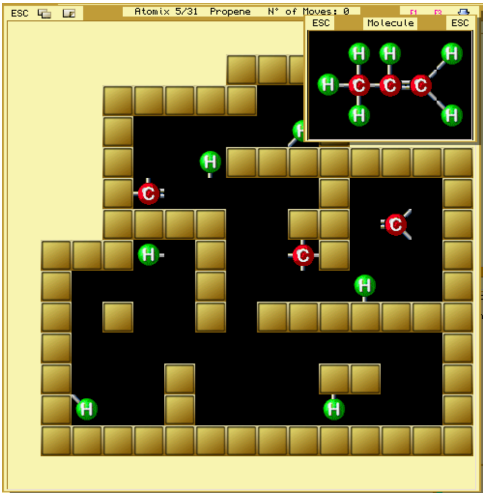
Opening screen of txtElite for QL

Atomics

Wolfgang Lenerz has released Atomics, a Katomics type game. This needs extended colours and at least a 512*512 screen. It's a graphical game where you try to build molecules out of different atoms by sliding things around.

The source files are also available from the author's website.

Download Atomics from http://www.wlenerz.com/qlstuff/

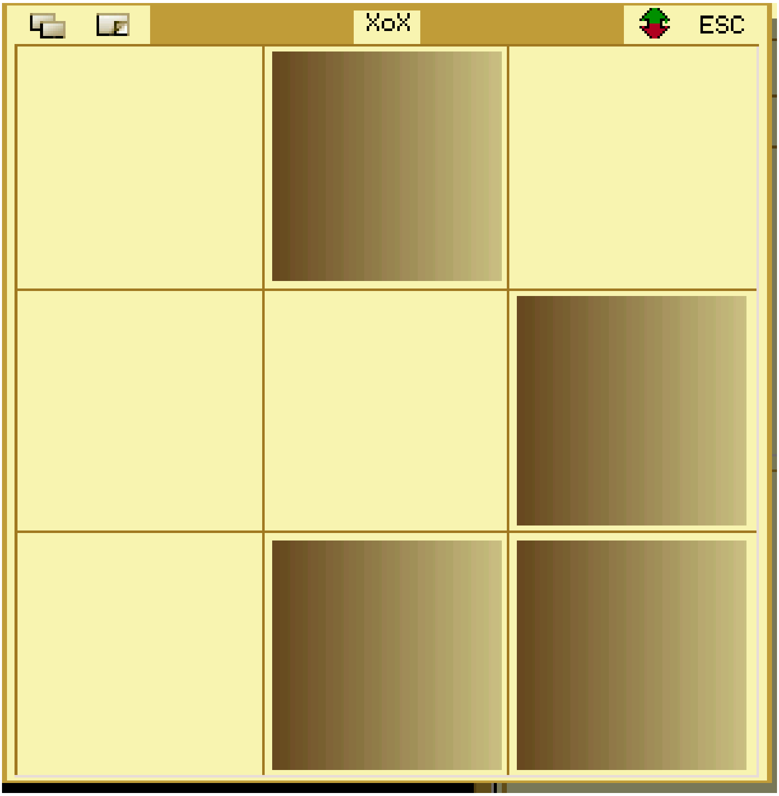


Atomics (picture from Wolfgang Lenerz)

XoX

Wolfgang has also released XoX, what he describes as “a silly little puzzle” which “might amuse small children”. XoX is a mini puzzle. There are nine tiles, which can change colour between the background colour and the tile colour. The purpose of the game is to have all nine tiles of the puzzle in the tile colour, EXCEPT for the tile in the middle.   You click on a tile to make it change its colour and back. However, clicking on a tile will also change (some of) its neighbouring tiles.

XoX needs the Menu Extension and a high colour screen. Wolfgang says: “The colour sprite is generated on the fly and someone might be interested in how it's done.”



XoX (picture from Wolfgang Lenerz)

QUO VADIS DESIGN

Independent Information
Technology Services

www.ql-qvd.com

QL/QDOS/SMSQ/E Software

QUO VADIS DESIGN Independent Information Technology Services

QL/QDOS/SMSQ/E Software

Welcome

Quo Vadis Design sells software for the Sinclair Quantum Leap computer (QL) and variants including a new OS called SMSQE.

The QL is a computer in its 25th year Anniversary.

Software emulations of the QL now exist which can run on a PC/Mac with Windows/Linux or Mac Operating systems.

News

QVD QL News Blog - keep up to date
News Blog
24/02/2009

Quo Vadis Design Website Launched
01/02/2009

FEATURED PRODUCT

Copyright © 2009 Quo Vadis Design. All Rights Reserved.

Home | Products | Support | Company | Contact | Privacy

QPC2 IS NOW FREE AND AVAILABLE FROM MARCEL KILGUS WEBSITE

http://www.kilgus.net/qpc/index.html

Congratulations to Marcel on becoming a Father

Check the QL News Blog on our website for updates.
www.ql-qvd.com/blog

Bruce@ql-qvd.com

Quo Vadis Design
38 Derham Gardens
Upminster
RM14 3HA
UK

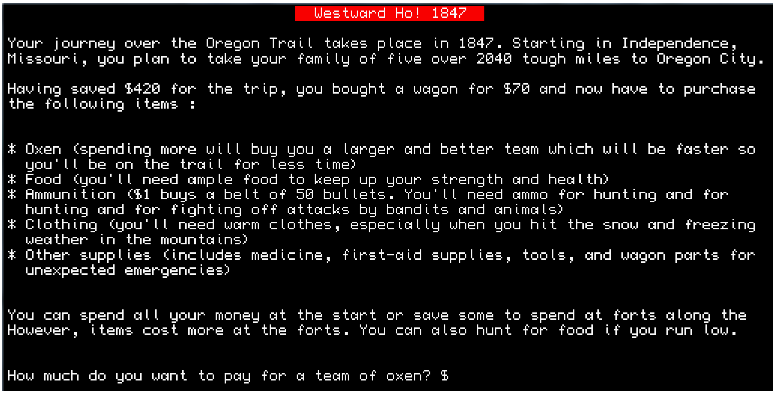
Tel: +44 (0)20 71930539
Fax: +44 (0)870 0568755

Westward-Ho!

Westward Ho is a QL port in SuperBASIC of a 1986 educational role-playing game in turn based on the well-known 1970s game The Oregon Trail, designed to teach schoolchildren about the realities of 19th century pioneer life on The Oregon Trail.
The player assumes the role of a wagon leader guiding his family of five settlers from Independence, Missouri, to Oregon's Willamette Valley over the Oregon Trail via a Conestoga wagon in 1848.

You start off with a fixed sum of money which must be spent carefully at forts en-route on food, ammunition, medicine, etc. All sorts of hazards and illnesses face you, and you must decide whether to hunt, run from hostiles approaching, etc. You must make good decisions, spend your money wisely, not waste too much time hunting and encounter some good luck to join the few who made it safely as settlers to the destination!

The game was ported by David Denham and is available to download from the Games page on my website at :
http://www.dilwyn.me.uk/games/index.html



```
                         Westward Ho! 1847

Your journey over the Oregon Trail takes place in 1847. Starting in Independence,
Missouri, you plan to take your family of five over 2040 tough miles to Oregon City.

Having saved $420 for the trip, you bought a wagon for $70 and now have to purchase
the following items :


* Oxen (spending more will buy you a larger and better team which will be faster so
  you'll be on the trail for less time)
* Food (you'll need ample food to keep up your strength and health)
* Ammunition ($1 buys a belt of 50 bullets. You'll need ammo for hunting and for
  hunting and for fighting off attacks by bandits and animals)
* Clothing (you'll need warm clothes, especially when you hit the snow and freezing
  weather in the mountains)
* Other supplies (includes medicine, first-aid supplies, tools, and wagon parts for
  unexpected emergencies)


You can spend all your money at the start or save some to spend at forts along the
However, items cost more at the forts. You can also hunt for food if you run low.


How much do you want to pay for a team of oxen? $
```

Screen shot of The Oregon Trail

QL Game Box Sets

RWAP Software have now released four in a series of box-sets of QL Software, to mark the 30th anniversary of the Sinclair QL. Included in the box sets are some games which are rarely seen in the QL community and have been resurrected as a direct result of the QL Software Preservation Project.

The QL has never really been thought of as a games machine, even though many good quality games have been released for it over the years, so this is quite a radical idea from RWAP. Whether you’re into games or not, these packs represent excellent value, and will definitely appeal to collectors of QL software.

The box sets are:

QL Adventures 1:
Return to Eden, Nemesis MKII, The Darkside of the Moon, The Lost Kingdom of Zkul, Horrorday

QL Adventures 2:
The Prawn, West, Mortville Manor, Aquanaut 471

QL Games 1:
Nebula II (both French and English versions), Hoverzone, Alien Hijack, FlightDeck, Black Knight, Puzzle Mania

QL Games 2:
QL Flight Simulator, Blast Buggy, Night Nurse, The Lonely Joker, Hopper, Gwendoline

More details and ordering appear on SellMyRetro:
http://www.sellmyretro.com/search/naturalSearch?keyword=ql+box-set
I have already heard some very positive feedback from early users who say that these were delivered quickly and that the standard of

presentation was very high (including the professionally designed covers). I'm not an adventures player so I won't be buying those, although I may well be buying one of the other two collections when I've decided which! Given the number of games in each box-set they are good value at £20 per set (or just £15 for Adventures 2) through SellMyRetro.com.

Game Box-Set 1 and 2 from RWAP Software (pictures from SellMyRetro.com)

Adventures Box-Set 1 and 2 from RWAP Software (pictures from SellMyRetro.com)

QUANTA Helplines Online

Past QUANTA Helpline articles are now online on the QUANTA website. They can be found on the "Library & Helpline" tab at the top of the site, just click on the HELPLINE sub-menu of the QUANTA website at http://www.quanta.org.uk or go direct to https://quanta.org.uk/software-library/helpline/
This takes you to a page with a list of the articles, where you can click on a link for the article concerned to view it as a PDF file.

Config Manual

Wolfgang Lenerz writes:
"I put up on the smsqe website (additional info and data section) the new manual about config levels 1 and 2."

To download it, go to this page:
http://www.wlenerz.com/smsqe/
go to the bottom of the page and click on the Additional Information and Data link, which takes you to a page where you can find information about Config Level 2 IDs and Manual.

Pointer Environment v2.05

Version 2.05 of ptr_gen (the pointer interface) and its associated files wman (Window Manager v2.07) and hot_rext (Hotkey System II v2.31) are now available to download from:

http://www.dilwyn.me.uk/pe/index.html or from :

http://www.wlenerz.com/smsqe/

(click on the Additional Information and Data link at the bottom of Wolfgang's page then you can download the files where it says Pointer Environment at the bottom).

Hot_rext is available in English, French and German versions. Note that the more recent versions of ptr_gen and wman are not likely to work on early Sinclair ROMs. Current versions will require at least a JS ROM, or Minerva. If you have a QL with AH or JM ROM version, stick to using ptr_gen v2.01 or earlier. Sadly, as far as I know, there is no document available listing the changes in each version.

DIY RAM Expansion

In the last issue I reported on a number of homebrew hardware products being worked on. Phill Harvey-Smith had been working on a RAM extension capable of going beyond the normal QL RAM expansion design limits of 640K. Here's an update he wrote since then:

"*I've now built the second version of the internal RAM board which allows me to expand beyond 640K by controlling the SP0..SP3 lines on the expansion connector (I've cut their connection to ground and fed them to a set of DIP switches on my board).*

This works and I can allow RAM in the expansion area and still allow peripheral boards to map themselves at the correct address, I've verified this with a little ROM that I created that prints it's load address in it's 'sign on' banner.

However one thing I have just noticed....with Minerva I can map RAM all the way to the top of RAM -16K to give me 880K (I actually set it for 864K, as my Sandy SuperDisk needs 2 ROM banks.).

"However using one of the original OS ROMS : AH, JM, JS, MGx

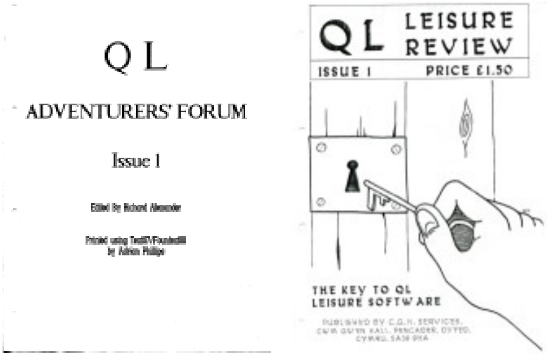
etc. the RAM is tested in 64K blocks so has to end on a 64K boundary, which means that the maximum RAM that can be achieved, and still use the SP0..SP3 to decode peripherals is 832K, basically SP0 and SP1 must be 0.

The trump card gets round this by cheating and moving it's ROM and I/O down into the I/O area below the beginning of RAM, then filling all the upper blocks with RAM which ends up being on a 64K boundary. This may be significant to anyone wanting to use a similar method for extending the RAM.”

New Magazines Page

I’ve added a new page to my website where I hope to include scanned copies of QL-related magazines. To start things off, I’ve included copies of the old CGH games magazines, the QL Adventurers Forum (all 9 issues) and QL Leisure Review (all 2 issues). These are downloaded as zipped collections of scanned pages (thanks to Peter Scott), one jpeg file per page, or as PDF files thanks to Derek Stewart.

Download from:
http://www.dilwyn.me.uk/mags/index.html



QL

ADVENTURERS' FORUM

Issue 1

Edited By Richard Alexander

Printed using Text87/Fountext88
by Adrian Phillips

QL LEISURE REVIEW

ISSUE 1 PRICE £1.50

THE KEY TO QL
LEISURE SOFTWARE

PUBLISHED BY C.G.H. SERVICES,
CWM GWYN HALL, PENCADER, DYFED,
CYMRU. SA39 9HA

QL Adventurers Forum and QL Leisure Review magazines

International QL Report Index

Timothy Swenson has compiled a spreadsheet containing an index to IQLR magazine. The index is supplied in Excel (IQLR_Index.xls) and Abacus (IQLR_aba) formats. Download the index from :
http://www.dilwyn.me.uk/docs/mags/index.html

QL World Listings List

A complete list of SuperBASIC program listings plus a few assembler listings from the QL User and QL World magazines is now available. Download it as a PDF file from http://www.dilwyn.me.uk/docs/mags/index.html

More Replacement Manuals

Replacement manuals for CST Q-488 IEEE interface and QSound/QPrint from ABC Electronik are now available from the Replacement Manuals page at :
http://www.dilwyn.me.uk/docs/manuals/index.html

MDI and FDI Drivers Update

Martin Head has sent me update files for his MDI and FDI drivers, which are now available to download from :
http://www.dilwyn.me.uk/utils/index.html

The updates include MDV driver v1.04, FDI driver v1.03 and a new ROM image (xDIRom) which is a ROM image with both the Microdrive and Floppy drivers combined. The drivers are now ROMable, and 16K ROM images are supplied for both the Microdrive (MDI), and Floppy (FDI) drivers. Commands have been added to the drivers to create blank Microdrive and Floppy images, without using separate SuperBASIC programs.

RWAP Software

We have been working hard to safeguard the future of the Sinclair QL through creating an online QL Wiki backed by a repository of QL titles.

The QL Wiki enables people to see what software and hardware was produced for the Sinclair QL. The repository enables people to purchase working copies when their microdrive cartridges fail, or they move onto disks or emulators. Where we can, we obtain permission from the copyright holders to re-release items on the market.

This has enabled us to bring some of the titles back to the market as well as re-kindle interest in the Sinclair QL and its emulators.

Old favourites brought back to life include:

E.V.A., Vroom
Black Knight, Double Block
Jungle Eddi, The Lost Pharaoh
Pengi, the Lonely Joker
Microdeal's Flight Simulator, QL Hopper, The King,
Night Nurse and Cuthbert in Space
Mortville Manor, Darkside of the Moon (£10)

We still retain our existing catalogue, including:

Open Golf, Return to Eden,
Stone Raider II, The Prawn,
Hoverzone, Lost Kingdom of Zkul,
Deathstrike, Horrorday,
Flightdeck and QWord

All games are available on disk or for use with Q-emuLator on the PC **from ONLY £5 each**.

Microdrive versions also available - from £10 each.
Memory / system limits may apply - please check before ordering.

Visit the QL Wiki for more details on software, books and hardware for the Sinclair QL:
www.rwapadventures.com/ql_wiki

RWAP Software
3 Dale View Court, Fulford, Stoke-on-Trent
Staffordshire ST11 9BA
Tel: 01782 398143
Email: rich@rwapservices.co.uk

Website:
www.rwapsoftware.co.uk or www.sellmyretro.com

Vroom by Pyramide

E.V.A by Westway

Mortville Manor by Pyramide

Double Block by Francois Lanciault

Night Nurse by Microdeal

Working to safeguard the QL for the future. www.rwapsoftware.co.uk

Some commands have been added to the Microdrive driver, to convert Martin's MDI images to and from QLay and Qemulator image file formats. The two update files are just replacement Readme's and code files for the two drivers already on the website.

BASIC is 50

The BASIC programming language is 50 years old this year – it derives from Dartmouth BASIC:

"Dartmouth BASIC is the original version of the BASIC programming language. It is so named because it was designed and implemented at Dartmouth College. The language was designed by John Kemeny and Thomas Kurtz as part of the Dartmouth Time Sharing System (DTSS) and was one of the first programming languages intended to be used interactively."
The above is a quote from the article on Wikipedia at:
http://en.wikipedia.org/wiki/Dartmouth_BASIC

Another article of interest at :
http://en.wikipedia.org/wiki/BASIC_language

A PDF of the manual for the Dartmouth BASIC from 1964 is available to view at :
http://web.archive.org/web/20120716185629/http://www.bitsavers.org/pdf/dartmouth/BASIC_Oct64.pdf

I just found this very interesting, since we are celebrating 30 years of the QL this year and SuperBASIC is such an integral part of the QL, so to find that the BASIC language is 50 years old this year as well is quite interesting.

QL Is 30

And speaking of 30 years of the QL, remember that the QL Is 30 event website is still available at :
http://www.gwicks.net/QLis30.htm

to give information about this event being held in Edinburgh in October. The QL Is 30 workshop will be held in the Gyle Hall of the St.Thomas's Episcopal Church, 75 -79 Glasgow Road, Edinburgh, EH12 8LJ from 10.00 to 17.00 on Saturday 11th October 2014. The event is being organised by the Scottish Users' QL Group (SQLUG) assisted by Just Words!

The St. Thomas's Episcopal Church is on the main road from Edinburgh City Centre to Edinburgh Airport. It is approximately 6km from the centre of Edinburgh and 4km from Edinburgh Airport.

Representatives of Quanta will be there (but don't let that put you off!) plus many of the leading lights of the QL community, including several from abroad.

RWAP's Sinclair QL Summer Stock Clearance

RWAP are having a Summer Stock Clearance - Sinclair QL Home Computers for £40 and bare Sinclair QL Microdrive units for £10 each – see :
http://www.sellmyretro.com/store/RWAP_Software_-_Retro_computer_specialists/All+categories/Retro+Computers/Sinclair/Sinclair+QL/Hardware

The QL computers are tested to ensure that they start up, but supplied without leads and/or power supplies – Rich Mellor says "we have not tested the keyboard or the Microdrive units.

The bare Microdrive units are untested, but from the stock we use ourselves to repair our QLs, so we expect them to be working."
If you need a printed QL User Guide – these are available for £5.

Microdrive Cartridges
Rich Mellor writes:
We have acquired some new old stock Microdrive cartridges:
http://www.sellmyretro.com/offer/details/New-Old-Stock-Sinclair-ZX-~~-QL-Microdrive-Cartridges-3763

and have some second hand Microdrive cartridges which we are offering in bundles of 12, ready for re-felting and re-use:
http://www.sellmyretro.com/search?keyword=microdrive+cartridges+lot&phraseType=allWords

The proceeds from the latter go to help fund the QL Commercial Software Preservation Project which has already succeeded in getting a range of games and utilities released as freeware (and now added to the Sinclair QL Homepage) – read more about it at :

http://qlforum.co.uk/viewforum.php?f=3&start=25
and download the software from the Sinclair QL Homepage:
http://www.dilwyn.me.uk/downloads.html .

We are still working our way through 100s of titles which need preserving / testing and adding to the QL Wiki – so if you can lend a hand, please get in touch! There are lots of personalities out there we still need to contact as they hold the copyright to this software, but we need your help to track them down and get in contact – after all, we don't get paid for this !!
The QL Wiki appears at:
http://www.rwapadventures.com/ql_wiki

SMSQmulator 2.03

Version 2.03 of SMSQmulator is now available. The emulator is now available in versions for Java 6, 7, or 8, depending on which is installed on your system, although the author Wolfgang Lenerz writes “Sooner or later I'm going to drop the support for java 6. Please upgrade your java version!”

There are various bug fixes, plus changes to handling of some key combinations such as better handling of CTRL-SPACE under Windows: F12 no longer needed, and an alternative mapping of CTRL-F5 to Scroll Lock, Ctrl-Space to pause/break.
Get the emulator from http://www.wlenerz.com/SMSQmulator/

QL-SD

After a summer break, QL-SD is available once more from Paul Veltjens through SellMyRetro.com, at :
http://www.sellmyretro.com/offer/details/QL-SD-internal-SDHC-Card-Interface-for-QL-3703 price £60 plus postage.

CHAIRMAN’S NOTES SARAH GILPIN

The celebration of QL is 30 in Edinburgh is fast approaching. I hope that as many members who can, will attend what promises to be an interesting day. I understand that some of our further flung members could be attending by electronic means.

Our thanks go to the Scottish Sub-group for all their hard work to put this event on, and also to Geoff Wicks for his help. We wish them all the best in having a successful day.

We look forward to more information regarding the day and also on the plans for the dinner. This is now urgently required to enable people to make arrangements to attend.

Although there will not be an official QUANTA presence, a number of the Committee will be coming as individuals. NEMQLUG sub-group will be there as this sub-group has been working hard to bring something of interest to the day. NEMQLUG is 21 this year and many of our original members are still regular attendees, so we will use the Edinburgh event as a place to celebrate our coming of age.

Are any of the other QUANTA sub-groups going to attend and bring some projects that they have been working on?

I look forward to seeing you there.

PARAMETER LISTS GEORGE GWILT

The effect of QL executable programs can be modified by means of a parameter list appended to the EXEC command when the program is run. I amended one of my programs so often, by adding yet another possible parameter to the list, that I produced a way of adding even more without the need for so much additional coding.

Since others may find this useful, I will describe the method here. First, I must explain that the parameters I am using are all stored in the program's data space. Also, I arrange in my program that

A6 always points to the start of that data space. This is achieved by the instruction:

```
     lea         (a6,a4.l),a6
```

issued near the start.

If the offset of a parameter from the start of data space is k, then the parameter is accessible by the address k(a6). If the program is assembled by GWASS, the RS instructions make it easy to set the offsets for the parameters. The RS instructions include RSSET, RS.B, RS.W and RS.L. The first of these sets the initial offset, usually 0, and the remainder allocate space. Thus:

```
     LAB1      RS.W       6
```

will assign the current RS value to LAB1 and increase the RS value by 6*2 bytes. A parameter list attached to the EXEC command must contain, for each parameter it is intended to alter, a code indicating which parameter is involved and its new value. The method shown here assumes that each parameter is indicated by a letter or number preceded by an asterisk.

The value for the parameter immediately follows the indicator. Thus *c45 will set the parameter indicated by "c" to 45. The subroutine par_lst, given below, searches through the parameter list processing all the parameters it finds.

To do this it has to know the number of possible indicators (given by the constant 'possn'), the possible indicators themselves, given in a list of bytes at 'poss' and a list of programs, one for each parameter, which will process the parameter. This last is given as a set of word offsets from the start of the list at 'do_tab'. Before par_lst is called A7 must be set to the start of the parameter list.

Listing of par_lst

```
par_lst   lea       4(sp),a3      par list
          move.w    (a3)+,d0      length of
                                  pars
          beq       pl9           none
          bra       pl1
pl2       move.b    (a3)+,d2
          cmpi.b    #'*',d2
          beq       pl3
pl1       dbf       d0,pl2
pl9       moveq     #0,d0     OK exit
pl8       rts


pl_er     moveq     #-1,d0
          bra       pl8
pl3       subq.w    #1,d0         adjust count
```

```
                                  for '*'
          bmi       pl8           error - no
                                  indicator

          move.b    (a3)+,d2  par type
          lea       poss,a0   list of
                              possibles

          moveq     #possn,d5 number of
                              possibles
                              less 1
pl7       cmp.b     (a0)+,d2  found? ..
          beq       pl6           .. yes
          dbf       d5,pl7
          bra       pl1

; D5 gives the parameter type number

pl6       lea       do_tab,a0     list of
```

```
                                        actions

          adda.w    (a0,d5.w*2),a0
                                        pick the
                                        required
                                        action ..
          jmp       (a0)                    .. and do
it
```

The parameters in my program happen to be all word integers. This meant that all the actions in do_tab are identical, apart from setting the address to which the parameter is to be set. Thus each action simply sets that address and then jumps to code, at do_a1, listed below, which translates the ASCII integer to binary.

The routine do_a1 requires that a_st point to the end of a space of at least 2 bytes in data space. It is perhaps worth noting that the vector cn_dtoi converts ASCII characters until the end of the buffer or until it finds an invalid character if earlier.

Listing of do_a1

```
; Note all actions must:
```

```
;          1. return to pl1
;          2. update D0 and A3
; On entry A4 is the address of the
; parameter relative to A6.

do_a1      movea.w   cn_dtoi,a2

           lea       a_st,a1       maths stack

                                   (rel to A6)

           movea.l   a3,a0         -> start of

                                   ASCII . .

           suba.l    a6,a0         --> .. rel to

                                   A6

           move.l    a0,d7

           move.l    d7,d6

           ext.l     d0

           add.l     d0,d7         --> end of

                                   buffer rel to

                                   (A6)

           move.l    d0,-(sp)

           jsr       (a2)          convert to
```

```
                        integer

bne       pl_er             --> error

move.l    (sp)+,d0      restore

                        count

sub.l     a0,d6

add.w     d6,d0         adjust

                        count ..

suba.w    d6,a3         .. and par

                        pointer

move.w    (a6,a1.l),(a6,a4.l)

bra       pl1
```

The two routines listed above are needed how ever many, or few, parameters there may be. But the more parameters there are the more tedious the writing of the actions for do_it becomes. My method of reducing the work is to use macros.

These are dota, dot, dote and dotz, described below. These macros also automatically set the correct value to possn and produce the lists at poss and do_tab.

To use the routine par_lst with 10 parameters, we would first set the parameter addresses.

```
rsset     0

par1     rs.w      1

par2     rs.w      1

...

par10    rs.w      1
```

Then we would need a_st pointing, relative to A6, to the base of the arithmetic stack. A stack of size 8, say, could be set in data space by:

```
a_st1    rs.b      8


a_st     equ       a_st1+8
```

The macros dota, dot and dot_e set up a table of entries to routines which process each parameter and place its value in data space.

The macro dotz sets the entry code for each parameter.
Thus, to be able to enter the 10 parameters, par1 to par10 with entry characters "a" and "j":

```
        dota

         dot       a
         dot       b
...
         dot       i
         dot_e


         dotz      a,par1
         dotz      b,par2
...
         dotz      i,par10
```

Listing of dot macros

```
dota      macro
possn     equ       _n-1
_n        set       0
do_tab
          endm
dot     macro
          dc.w      do_\1-do_tab
          if _n=0
_l       set       ?"\1"
          else
_l       set       ?"\1|_l~"
          endif
_n        set       _n+1
          endm


dot_e     macro
poss      dc.b      _l
          endm


dotz      macro     item,name
do_\1     lea       \2,a4
          bra       do_a1
          endm
```

The first two instructions in the macro dota may appear confusing. You may wonder why possn is set equal to one less than a variable which is defined as zero in the following instruction. The reason is that when an assembler, such as GWASS, comes to the former instruction it can't assemble it completely because, at that time, the variable, in this case _n, is undefined.

When the complete program has been scanned _n will have been given a value, first by the second of the two instructions and later by the dot macros. At this stage the assembler will go back to all the incomplete instructions inserting values for any previously undefined variables.

QUANTA HELPLINE DILWYN JONES

Members wishing to submit helpline requests via email can use the email address helpline@quanta.org.uk or if you prefer to use traditional post, please send the helpline request to me via the address printed inside the front cover of the magazine.

Obviously, we cannot guarantee to answer every query we receive, but we will do our best! Where we have been unable to answer the queries, we may print the help request as an open request in the magazine to ask if any of the readers can come up with a solution.

Of course, if readers feel that they have a better solution than we came up with, or would like to correct any errors we make, please write to us!

(In the last issue Dilwyn was discussing screen sizes and how to

get this information in to your program depending on MODE, we continue… Ed.)

MODE 4 and MODE 8: 8 pixels per two bytes.

```
    LET visible_length = SCR_XLIM DIV 4
```

256 colour (8-bit) screen mode 16: One byte per pixel

```
    LET visible_length = SCR_XLIM
```

16 bit colour modes 32 or 33 (QPC2, QXL, SMSQmulator, Q40, Q60): 2 bytes per pixel

```
    LET visible_length = 2*SCR_XLIM
```

To decide which of those to use, we can use the DISP_TYPE function to tell us the screen display mode:

```
    LET display_mode = DISP_TYPE
    SELect ON display_mode
      =0,4,8 : LET visible_length = SCR_XLIM DIV
    4
      =16    : LET visible_length = SCR_XLIM
      =32,33 : LET visible_length = 2*SCR_XLIM
    END SELect
```

If new display modes are introduced in the future, it is simple enough to add to this once you know how many bytes per pixel. For example, a “true colour” 24-bit mode (MODE 64) would use 3 or 4 bytes per pixel (most probably a long word per pixel – 4 bytes).

So now we know how much of each Aurora video line to save, we can create a program which will step through the video memory a line at a time, saving just the part of each line needed:

```
OPEN_NEW #3,filename$
REMark set visible_length as above…
LET video_address = SCR_BASE
FOR y = 0 TO SCR_YLIM-1
 PRINT#3,PEEK$(video_address,visible_length);
 LET video_address=video_address+SCR_LLEN
END FOR y
CLOSE #3
```

Please note that the line starting with PRINT #3 has a semi-colon at the end. This is to prevent it adding an unwanted linefeed character after the PEEK$ function.

Things start getting more complex when we need to read this back, as we need to read a line at a time from the file and poke it into the screen in the right place. Fiddly, but not impossible:

```
OPEN_IN #3,filename$
REMark set visible line length as above
...
LET video_address = SCR_BASE
FOR y = 0 TO SCR_YLIM-1
  LET lne$ = “” : REM fetch copy of current
line
  FOR x = 1 TO visible_length : lne$ =
lne$&INKEY$(#3)
```

```
REMark place the line into the screen memory
  POKE$ video_address,lne$
  REMark move pointer to start of next line
  LET video_address = video_address +
SCR_LLEN
END FOR y
CLOSE #3
```

This routine will be quite slow for large screens, as it reads the graphics data back byte by byte, line by line. It may be possible to speed it up by using a command such as INPUT$ from some toolkits (I think Turbo Toolkit and DJToolkit have such functions) which read a given number of bytes from a channel at a time, letting you replace the two lines LET lne$=”” and FOR x=1 TO

… with something like:

```
LET lne$=INPUT$(#3,visible_length)
```

When I tried this on my computer, the speedup was quite drastic compared to using INKEY$.

One word of warning: in the higher colour and resolution modes, saving screens can result in very large files. I was using 16-bit colour mode in 1024x768 pixel resolution and it created a graphics file 1,572,864 bytes long. Suddenly I realised why floppy disks can go out of fashion!

If you want to work out in advance what size a file will be, the basic calculations are as above. Work out how many bytes across a line for the mode you are using, multiply it by the depth of the screen in pixels and you have a length (in bytes) of the resultant screen.

A final word on this, for those who may want their software to also work on non-SMSQ/E systems where these extensions are not present. On the overwhelming majority of QDOS systems (with the exception of the uQLx and Atari emulator cards) the screens will always be 32,768 bytes long, starting at address 131072 since standard unpatched QDOS cannot provide for larger displays or more colours.

The question then is how to protect your program from errors caused by non-existent extensions. This is actually easier than you might think: check the ROM version and use IF statements to take different actions. Here’s a simplified routine, which may not work on the Aurora for the reasons described above:

```
v$ = VER$
REMark default values for QDOS
base=131072
size=32768
IF v$ = ‘HBA1’ THEN
  REMark aha, running on SBASIC
  base=SCR_BASE
  size=SCR_YLIM*SCR_LLEN
END IF
SBYTES filename$, base, size
```

Another way is to make use of freely available extensions in toolkits which work on both QDOS and SMSQ/E. One example is the Display-Code toolkit I wrote a few years ago, published in

QL Today at the time and now available to download as freeware from my website’s Toolkits Page at : http://www.dilwyn.me.uk/tk/index.html

QUANTA members can also use a set of extensions called Screen Parameters, written by Bruno Coativy, available on Library Disk UT01. This handy little package provides extensions to tell you the screen address, size, line length and so on.

Q. OK, I think I get that. Now, how can I cut and paste parts of the picture from one place on the screen to another?

A. Now it becomes really complex! If you are familiar with pointer environment programming software, it is best to use existing routines within the programming tools to do this, as the pointer environment provides routines to save and restore areas of the screen.

Not only does this prevent you having to write unnecessary code, it provides the most future-proof code basis to ensure that the program you wrote does not run into problems in the future when system designers bring in new video modes, for example.

Sometimes, though, your needs cannot be fully met by existing software, and anyway, we are all tinkerers, it is our right to explore, write code, peek and poke here and there – that’s what being a QL user is all about!

To offer a solution to this, I am going to print a slightly modified listing based on a routine from Tobias Fröschle, which he published on QL Forum. It involves working out the address of where the top left of the block to be copied starts, how many bytes wide it is, how many lines deep and from that how many bytes needed to store those lines.

To keep things as simple as possible, this routine does not try to cater for the Aurora special case described above, and it requires a command such as MOVE_MEMORY to move chunks of memory about. Such commands are commonly available in many toolkits, probably the best known being the Turbo Toolkit.

In modes 4 and 8, because there are multiple pixels per byte, it is easier to write these routines if you make sure that the block to be saved is a multiple of 8 pixels wide, and that the block starts on a word boundary (a multiple of 2 bytes or 8 pixels) across the screen. Otherwise you get into the realms of copying part bytes and shifting and rotating bits and pixels and that is simply beyond the scope of this article!

This version of the routine builds up an area save header similar to those used by the pointer environment, for three reasons:

1. To be compatible with pointer environment area save files
2. To allow the copied area to be saved to a file as a _pic file, one of the most common QL graphics file formats
3. To allow the copied area to be printed with the SDUMP command, by supplying the address value returned from this function to the SDUMP command

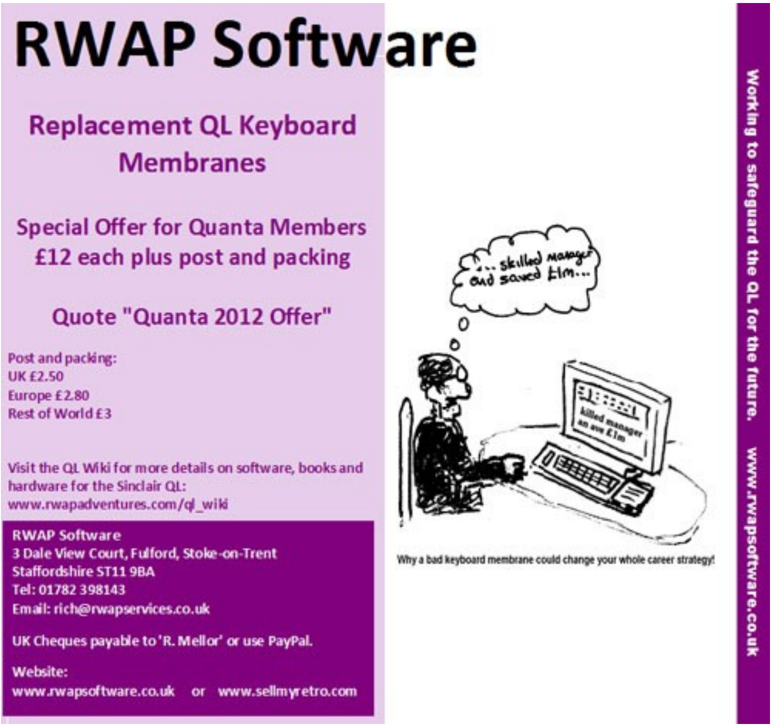
To call the routine, call the ScrSave function with four parameters representing the x and y co-ordinates of the top left of the part to be saved, plus the width and height (in pixels) of the area to be saved in memory or copied. In QL modes 4 and 8, the x and w values (origin across and width) **MUST** be multiples of 8 pixels wide.

start_address = ScrSave(x_origin,y_origin, wide, high)

The routines have no error trapping as they stand, to keep them fairly short for publication, I'm sure readers will have fun adding this.

```
1000 DEFine FuNction ScrSave (x, y, w, h)
1010   LOCal llen, scrBase, bpp, pixlen, blen,
maddress
1020   LOCal bstart, mneed, srcPtr, dstPtr,
lineNumber
1030   :
1040   REMark set defaults for QDOS in case no
SMSQ/E
1050   scrBase = 131072 : llen = 128 : scrMode
= 0
1060   bpp = (128 * 8 / 512) / 8 : REMark Bytes
per pixel QL modes (0.25)
```

```
1070   IF VER$ = "HBA" THEN
1080     REMark use SMSQ/E values
1090     scrBase = SCR_BASE
1100     llen = SCR_LLEN
1110     scrMode = DISP_TYPE
1120     SELect ON scrMode
1130       =0,4,8 : bpp = (llen * 8 /
SCR_XLIM) / 8 : REMark QL modes
```



```
1140       =16    : bpp = 1 : REMark 8-bit
colour mode
1150       =32,33 : bpp = 2 : REMark 16-bit
colour mode
1160     END SELect
1170   END IF
1180   :
1190   REMark length of one window scanline in
bytes, rounded up
1200   REMark in QL colour modes where there is
more than one
1210   REMark pixel per byte
1220   IF scrMode < 16 THEN
1230     blen = bpp * 8 * INT ((w + 7) / 8)
1240   ELSE
1250     blen = bpp * w : REMark higher colour
modes
1260   END IF
1270   :
```

```
1280   REMark start address within screen
1290   bstart = scrBase + (y * llen) + (x * bpp)
1300   :
1310   REMark memory needed +5 words for area save header
1320   mneed = blen * y + 10
1330   maddress = ALCHP(mneed)
1340   :
1350   REMark set up area save header before graphics
1360   POKE_W maddress, HEX('4afc'): REMark header flag 19196
1370   POKE_W maddress + 2, w      : REMark remember width
1380   POKE_W maddress + 4, h      : REMark remember height
1390   POKE_W maddress + 6,blen    : REMark remember line increment
```

```
1400   POKE maddress + 8,DISP_TYPE : REMark
remember mode number
1410   POKE maddress + 9,0         : REMark
zero the spare byte
1420   :
1430   dstPtr = maddress + 10 : REMark start
just after header
1440   srcPtr = bstart        : REMark start
point of first line in screen
1450   FOR lineNumber = 1 TO h
1460     MOVE_MEMORY srcPtr TO dstPtr, blen
1470     srcPtr = srcPtr + llen : REMark next
line down screen
1480     dstPtr = dstPtr + blen : REMark next
line store area
1490   END FOR lineNumber
1500   RETurn maddress
1510 END DEFine ScrSave
1520 :
```

It is possible to adapt the routine to allow the copied area of the screen to be saved to a file if you wish, a copy-to-file routine. The easiest way to do this is to make the ‘mneed’ variable global, by removing it from the LOCal list in line 1020.

That way, all you need to do is to use an SBYTES command to save the graphic to a file using the start address and length values:

```
        SBYTES filename$, start_address, mneed
```

This saves a standard QL _pic file. It can be reloaded later by allocating the same amount of memory as the length of the file, and will then be the same as a copied area from the routine above.

```
        start_address = ALCHP(FLEN(\filename$))
        LBYTES filename$, start_address
```

Slight digression: let me formally document the _pic file format here. It’s basically a block of graphics, with a 10 byte header preamble in the file storing a flag to identify the type of flag, the width and height of the picture (in pixels).

The line increment of the graphics in the file (bytes from start of one line to the next) and the mode number, plus a single spare unused byte. The offset and length below are in bytes from the start of the file:

Offset	Length	Function
0	2	Flag of $4AFC (decimal 19196)
2	2	Width, in pixels
4	2	Height, in pixels
6	2	Line increment, in bytes
8	1	Screen mode number
9	1	Spare, unused byte.
10		start of graphics, in same format as screen mode it was saved from.

Having copied the relevant part of the screen, you'll need a matching routine to paste it back somewhere else on the screen. Call this with the address returned by the routine above as the first parameter, to tell it where to paste FROM, then the x and y co-ordinate of the point on the screen the pasted image should appear at.

Note that for QL modes 4 and 8 the x co-ordinate should be a multiple of 8 pixels across, and the co-ordinates cover the whole screen, starting from top left, down to the bottom right.

Graphics should be pasted back to the same screen mode as that they were saved from, otherwise it will either look very odd, or possibly even crash the computer!

```
1530 DEFine PROCedure ScrPaste (srcAddr,x,y)

1540   REMark paste the part screen stored at

scrAddr

1550   REMark to the co-ordinates x,y (top
left)
1560   REMark in mode 4 or 8, x must be
multiple of 8
1570   LOCal scrBase, llen, blen, bpp
1580   LOCal srcPtr,dstPtr,lineNumber
1590   :
1600   REMark set defaults for QDOS in case no
SMSQ/E
1610   scrBase = 131072 : llen = 128
1620   bpp = (128 * 8 / 512) / 8 : REMark Bytes
per pixel for mode 4
1630   IF VER$ = "HBA" THEN
1640     REMark use SMSQ/E values
1650     scrBase = SCR_BASE
1660     llen = SCR_LLEN
1670     scrMode = DISP_TYPE
1680     SELect ON scrMode
1690       =0,4,8 : bpp = (llen * 8 /
SCR_XLIM) / 8 : REMark QL modes
```

```
1700       =16    : bpp = 1 : REMark 8-bit colour mode
1710       =32,33 : bpp = 2 : REMark 16-bit colour mode
1720     END SELect
1730   END IF
1740   :
1750   REMark length of a scan line
1760   blen = PEEK_W(srcAddr+6)
1770   :
1780   REMark where are we pasting from?
1790   srcPtr = srcAddr+10
1800   :
1810   REMark where are we pasting to?
1820   dstPtr = scrBase + (y * llen) + (x * bpp)
1830   :
1840   FOR lineNumber = 1 TO PEEK_W(srcAddr+4)
1850     MOVE_MEMORY srcPtr TO dstPtr, blen
```

```
1860     srcPtr = srcPtr + blen
1870     dstPtr = dstPtr + llen
1880   END FOR lineNumber
1890   :
1900   REMark to release heap memory
automatically after paste,
1910   REMark just remove the REMark from the
next line
1920   REMark RECHP srcAddr : srcAddr = 0
1930 END DEFine ScrPaste
```

As it stands, the ScrPaste procedure does not release the memory used to hold the copy in memory/ Line 1920 shows how to do this automatically after each paste – just remove the REMark statement before the RECHP srcAddr:srcAddr=0 statements.

Alternatively, if you think you might want to make multiple pastes of the same image, leave that as it stands and elsewhere in your program put the RECHP srcAddr:srcAddr=0 statements to release the heap memory once you have finished pasting.

OK, so now you know how to load and save screen and _pic files, plus how to copy and paste graphics, off you go and write a graphics program, please!

(OK I’m on it—Ed.)

NEMQLUG SUB-GROUP NEWS SARAH GILPIN

In August 1993 Steve Papierowski invited local QUANTA members to meet at New Moston Library to form the North East Manchester QL Sub-Group. Of the original members who attended and who still regularly come to NEMQLUG meetings are John and Alison Southern, Chris Grogan, Vince Karaba, Alex Wells and John and Sarah Gilpin.

Initially we held our meetings at New Moston library, with no charge from the library. When New Moston decided to charge us, it was by the hour and was beyond our means; so we transferred to Steve’s house. Unfortunately by then Steve was terminally ill so after a couple of meetings the group transferred to their current venue at 181, Urmston Lane, Stretford with an occasional foray to Poynton; chez Southern.

We still owe Steve our thanks for setting up the group.

Over the years the sub-group has held workshops and hosted QUANTA AGM’s, which we have all enjoyed. Each year the sub-group hold a dinner somewhere instead of a regular meeting; usually at the local Chinese eatery by popular demand.

We regularly have seven or more attending our monthly meetings, held on the last Thursday of the month (except for December) and everyone is welcome from 7.30pm onwards.

All this means that

NEMQLUG is 21

QL IS 30 WORKSHOP — GEOFF WICKS

The QLis30 workshop dinner will be held on Friday 10th October in the Holiday Inn, 132 Corstorphine Road, Edinburgh, EH12 6UA.

The Holiday Inn is about 2 km from the QLis30 workshop venue in the direction of the city centre. It is near the Edinburgh Zoo and is served by the airport bus. www.google.co.uk/maps/@55.9431159,-3.266964,16z

The room is booked from 19.00 to 23.00 and the final menu is still to be decided, but it will have a Scottish theme.

SPECIAL DIETARY NEEDS

Please inform us of any special dietary needs, including vegetarian, when booking as we have to inform the Holiday Inn in advance.

The cost is £25 for a three course meal plus coffee. As the Holiday Inn requires payment in full, a month in advance, all bookings must be in and fully paid by the end of August.

PAYMENT METHODS

For obvious security reasons we are unable to put bank account and personal contact details online and would ask you to use one of the following procedures.

1: Payment by cheque for £25 made out to SQLUG and sent to John Sadler. For John's contact details please use the email link on the SQLUG website:

www.jms1.supanet.com/SQLUG/intro.htm

2: Payment by PayPaL. QUANTA has kindly agreed to allow payment via their PayPal account. This may be the cheapest way for non-UK-ers to pay.

There is a charge for using PayPal and, for foreign currency users, exchange and transfer costs. UK users should pay £26 and non UK users £28 to cover these charges.

For details of how to pay via the PayPal website please contact QUANTA's treasurer via their website link:

www.quanta.org.uk

If paying by any form of transfer please ensure that QLis30 and your name are included in the payment details.

(Note: John Gilpin adds "*QUANTA has added an extra page to it's Web Site containing two new PayPal Payment buttons which can be used to pay for the dinner at the QL is 30 event.*

In an effort to limit the possibility of misuse by the general public who have general access to this site, this new page has not been added to the navigation details and will therefore be difficult for anyone to find without the following info:-

Please enter (or Copy) the FULL URL into your browser. The direct link for payment via PayPal is : https://quanta.org.uk/ql-30/" Ed.)

COMMENT — LEE PRIVETT

In the last issue, a report regarding the QUANTA Committee Meeting immediately following the AGM included the cost per copy of the printed version of the QUANTA Magazine was discussed.

We are still waiting for members input on the questions asked:

1. At what point do we consider the cost per printed magazine to be uneconomic?

2. What solution should we take?
3. To consider the costs of purchasing our own colour laser printer, stapler and guillotine, and printing our own magazine.

OR

4. To consider making the QUANTA Magazine an eMag only?

How many members do not have ANY facilities to download and read a .pdf file? (please send your views to:- Lee Privett – Magazine Editor)

This will help us to make these decisions on behalf of the members.

LATE NEWS — DILWYN JONES

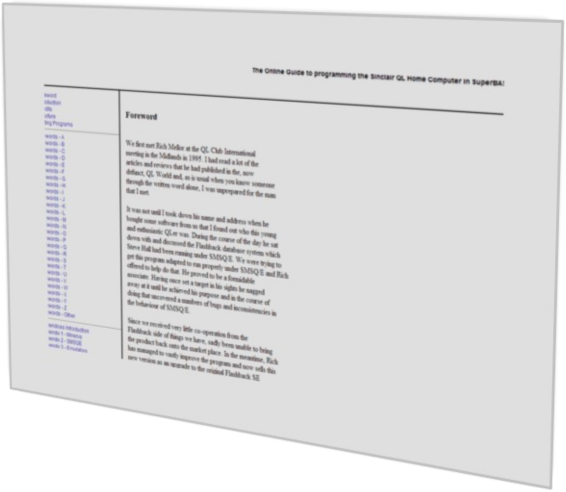
SBASIC / SuperBASIC Reference Manual - HTML

Thanks to some sterling work by Tomas Kral to further develop the automatic conversion tool used to convert the manual from Text 87 to HTML, the first draft version of the SBASIC/SuperBASIC Reference Manual is now online at: http://www.rwapsoftware.co.uk/SBASIC_reference_manual_online/Foreword.html

At the time of writing, volunteers are required, as Rich Mellor will not have time for this - it is now going to be a community project. If nobody comes forward to help with changes, then they are unlikely to be done!

Some of the changes for which help is needed:

a. Write a home page to form the main link to the manual
b. Improve CSS and layout
c. Some more work on the conversion tool to handle the non-English characters - especially needed for Appendix 8 and 9.
d. Volunteers to take ownership of a specific page and hand edit it to improve the layout - remove some of the line breaks for example in each paragraph.



The Online Guide to programming the Sinclair QL Home Computer in SuperBA

Foreword

We first met Rich Mellor at the QL Club International meeting in the Midlands in 1995. I had read a lot of the articles and reviews that he had published in the, now defunct, QL World and, as is usual when you know someone through the written word alone, I was unprepared for the man that I met.

It was not until I took down his name and address when he bought some software from us that I found out who this young and enthusiastic QLer was. During the course of the day he sat down with and discussed the Flashback database system which Steve Hall had been running under SMSQ/E. We were trying to get this program adapted to run properly under SMSQ/E and Rich offered to help do that. He proved to be a formidable associate. Having once set a target in his sights he nagged away at it until he achieved his purpose and in the course of doing that uncovered a numbers of bugs and inconsistencies in the behaviour of SMSQ/E.

Since we received very little co-operation from the Flashback side of things we have, sadly been unable to bring the product back onto the market place. In the meantime, Rich has managed to vastly improve the program and now sells this new version as an upgrade to the original Flashback SE

Screen shot of the introduction page

If you would like to offer to help, please contact Rich Mellor at RWAP Services.

COPY DATE for OCT/NOV 2014 ISSUE is: 5th OCT 2014

Please ensure that your copy reaches the editor on or before that date. Copy on Paper, Magnetic Media or by Email. Text & listings are preferred in ASCII format as text files, Microsoft™ Word Documents or in QL Quill formatted files. All graphic submissions in as higher resolution as possible, preferably in PNG format, other forms of submission can also be accepted, please contact the Editor for details.

All Contributions are Welcome

Copy by Email to - editor@quanta.org.uk

The online QL community

The_QL_Forum

The Sinclair QL Forum
A Place for Sinclair QL Users to Meet!

Search... Search
Advanced search

Board ind

QL-SD Fitted

18 posts • Page 1 of 2 • 1 2

Forum rules

POSTREPLY

Search this topic... Search

QL-SD Fitted
by XorA » Fri Mar 14, 2014 6:57 pm

Well I fitted my QL-SD, it is as predicted in manual unstable with SGC.

Looks like I need to get some expans

Here are the pics.

https://plus.google.com/102707

XorA
Bent Pin Expansion Port

Posts: 98
Joined: Thu Jun 02, 2011 11:31 am
Location: Glasgow, Scotland, UK

Re: QL-SD Fitted
by Paul » Fri Mar 14, 2014 9:33 pm

Thank you for posting this.
Do You feel I should add more warnings about GoldCard and SuperGoldCard?
If you like to get an image for 128K then just tell me.
Kind regards
Paul

Now with QUANTA section!

New online chat facility!

The Sinclair QL Forum
A Place for Sinclair Q

Search... Search
Advanced search

FAQ Register Login

WERNER by Karl Shaefer

Forum rules

POSTREPLY

Search this topic... Search

6 posts • Page 1 of 1

WERNER by Karl Shaefer
by RWAP » Fri Feb 14, 2014 7:05 am

Does anyone know anything about the game WERNER by Karl Shafer?

I came across this game - it seems to be a variant on Boulderdash -see screenshot.

RWAP
RWAP Master

Posts: 1100
Joined: Sun Nov 28, 2010 5:51 pm
Location: Stone, United Kingdom

Free to join!

No registration needed to view

Please stop by for a visit!

www.qlforum.co.uk